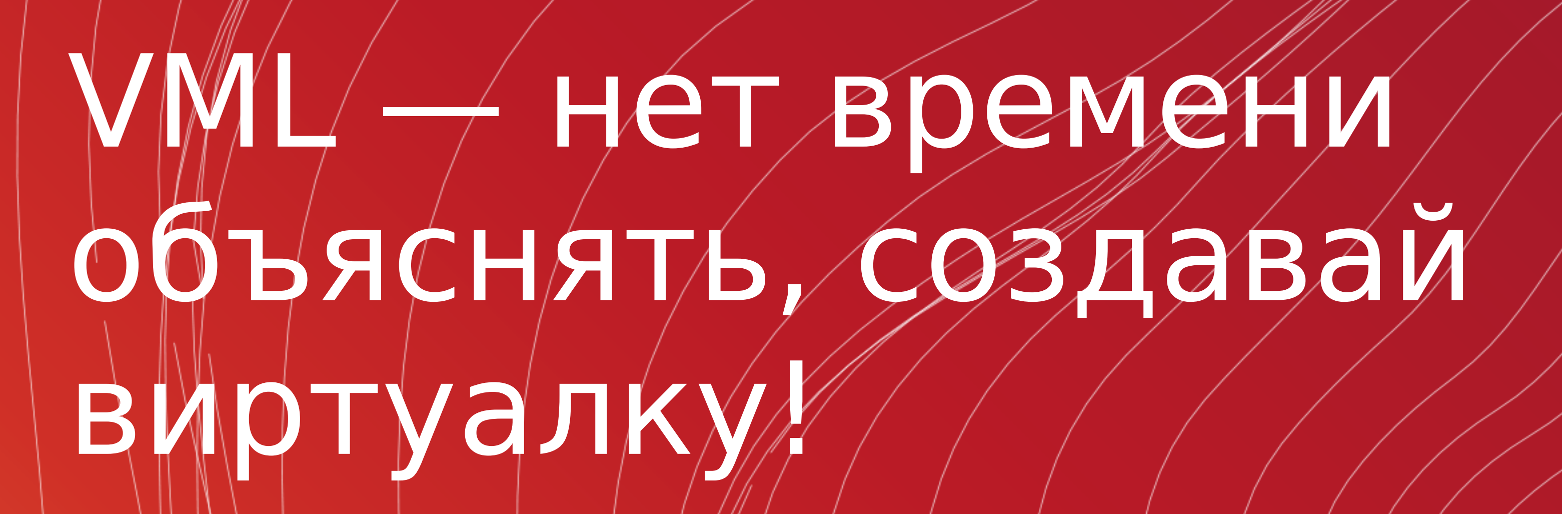




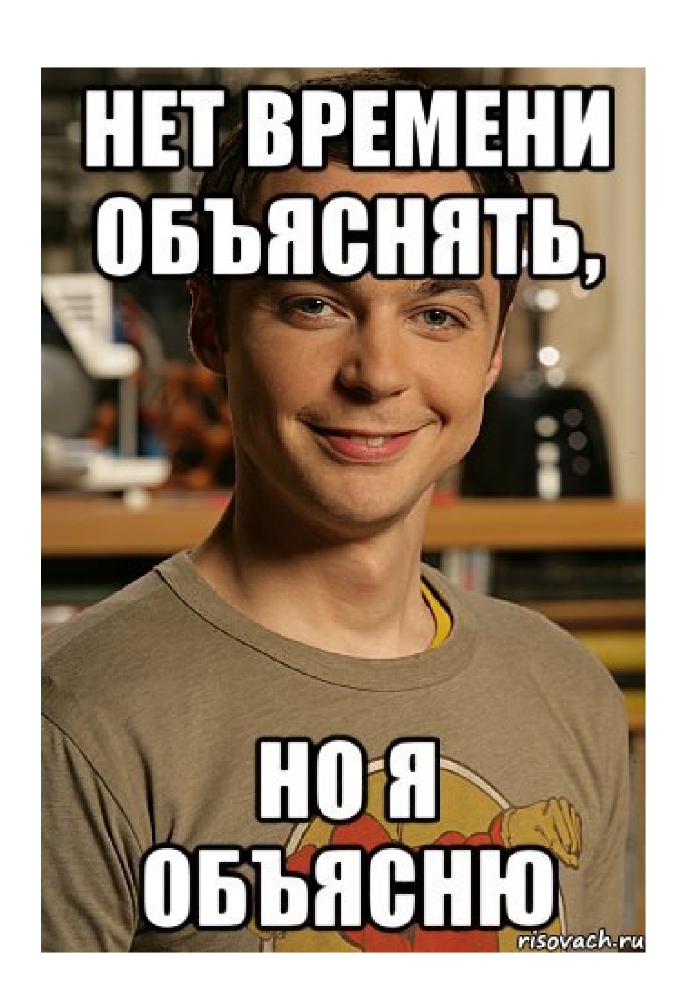
Руководитель отдела виртуализации, Базальт СПО



О, VML!

VML – это инструмент для простой и прозрачной работы с виртуальными машинами qemu.

VML может инициализировать образы с помощью cloud–init.





История создания

VML – это инструмент для простой и прозрачной работы с виртуальными машинами qemu.

VML может инициализировать образы с помощью cloud-init.

Для того чтобы программы было легко запускать на разных машинах, было решено написать программу на Rust. А ещё потому, что он модный и молодёжный.

Первая версия была написана на OcaML, но на нём меньше библиотек, и их нужно было собирать отдельно, а на Rust их можно завендорить.



Долгая история тестирования

- Компания «Базальт СПО» выпускает облачные образы и их требовалось тестировать. Изначально они тестировались в собственном облаке OpenStack, но со временем мы перестали его поддерживать.
- Некоторое время мы тестировали образы в облаке Mail,
но в итоге поняли, что быстрее всего тестировать всё локально, чтобы тесты могли запускаться сразу после сборки без копирования файлов по сети.



Зачем всё это?

VML можно использовать для разных задач:

- мы активно применяем его для тестирования облачных образов
- для создания мини-облаков, использования в CI
- VML используется для управления виртуальными машинами, на которых работает buildbot
- для разворачивания и тестирования кластера Kubernetes
- для проверки запуска и работы софта на разных дистрибутивах



Про cloud−init

Cloud−init − это программа для первоначальной «настройки» виртуальных машин, обычно применяющаяся в облачных платформах. Но использовать её можно и локально.

Distributions:

- Alpine Linux
- ALT Linux
- Arch Linux
- Debian
- DragonFlyBSD
- Fedora
- FreeBSD
- Gentoo Linux
- NetBSD
- OpenBSD
- Photon OS
- RHEL and Co
- SLES/openSUSE
- Ubuntu

Clouds:

- Amazon Web Services
- Microsoft Azure
- Google Cloud Platform
- Oracle Cloud Infrastructure
- OpenStack
- Softlayer
- Rackspace Public Cloud
- IBM Cloud
- DigitalOcean
- Bigstep
- Hetzner
- Joyent
- CloudSigma
- Alibaba Cloud
- OVH
- OpenNebula
- Exoscale
- Scaleway
- CloudStack
- AltCloud
- SmartOS
- UpCloud
- Vultr
- Zadara Edge Cloud Platform
- 3DS Outscale

Additionally:

- Bare metal installs
- LXD
- Qemu
- PVE
- Metal-as-a-Service (MAAS)
- VMware



Метаданные cloud−init

Cloud-init поддерживает разные типы метаданных. VML использует метаданные в формате nocloud. Это iso с несколькими файлами.

```
#cloud-config
users:
  - name: root
    ssh_authorized_keys:
      - <YOUR_PUBLIC_KEY>
```

```
cloud-localds seed.img userdata.yaml
```

```
qemu-system-x86_64 -m 1024 -net nic -net user \
    -hda ubuntu-20.04-server-cloudimg-amd64.img \
    -hdb seed.img
```



Как собрать на Rust?

VML поддерживает набор основных операций для работы с виртуальными машинами. Такие как запуск, остановка, копирование файлов и выполнение команд. Рассмотрим на примере сборки самого vml.

Создадим виртуальную машину:

```
vml run -i alt -m 4G --nproc 4 --ssh-user builder -n vml-builder
```

Установим необходимые пакеты:

```
vml ssh root@vml-builder -c 'apt-get update'
vml ssh root@vml-builder -c 'apt-get install -y rust-cargo gcc'
```

Скопируем исходный код:

```
vml ssh -n vml-builder -c 'mkdir src'
vml rsync-to -avP -n vml-builder -s ./ -d src/vml/
```

И соберем vml:

```
vml ssh -n vml-builder -c 'sh -c "cd src/vml; cargo build"'
```



Пример запуска

Посмотреть доступные образы можно командой:

```
vml image available
```

Чтобы запустить виртуальную машину, нужно ввести:

```
vml run -i alt-sisyphus --wait-ssh -n test
```

Где alt-sisyphus — название образа

Чтобы зайти в созданную виртуальную машину достаточно команды:

```
vml ssh -n test
```



С чем работает?





Сеть

Настрой-ка
мне сеть



Есть мааааленькая проблема
с настройкой сети.

В GNU/Linux для настройки сети требуются рутовые привилегии, поэтому vml использует только qemu user network или tap network.
Но его нужно будет предварительно настроить самостоятельно.



Ручная настройка сети

Пример настройки сети, при котором виртуальные машины могут взаимодействовать друг с другом.

```
ip link add br0 type bridge
ip link add dummy0 type dummy
sysctl net.ipv4.ip_forward=1
ip link set br0 up
ip link set dummy0 up
ip link set dummy0 master br0
for n in $(seq 0 1); do
    ip tuntap add dev "tap$n" mode tap
    ip link set "tap$n" master br0
    ip link set "tap$n" up
done
ip a add dev br0 172.16.0.1/24
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
iptables -t nat -A POSTROUTING -s 172.16.0.0/24 -j MASQUERADE
```



Ansible

Для того чтобы выполнять более сложную настройку виртуальных машин, можно использовать Ansible.

Для запуска Buildbot на двух нодах master и worker достаточно запустить команды:

```
vml run --net-tap tap0 --net-address 172.16.0.2/24 --net-gateway 172.16.0.1 bb/master

vml run bb/worker
```

После этого их можно будет настроить с помощью Ansible:

```
ansible-playbook -i ../../../files/scripts/inventory.py playbook.yaml
```

Так как VML поддерживает Ansible dynamic inventory.



Пример использования Ansible



Групповое управление виртуальными машинами

Виртуальные машины, созданные с помощью VML, можно объединять в группы, используя теги и префиксы.

Имя виртуальной машины состоит из нескольких частей, мы взяли за основу древовидную файловую систему с каталогами и именами файлов.

Большинство команд vml можно применить сразу к группе виртуальных машин.

Префиксы являются **одним из самых удобных способов** обозначить такую группу.

Например, команда `vml stop -p k8s` остановит все виртуальные машины, имя которых начинается с `k8s/`.



Kubernetes

```
vml run --net-tap tap1 --net-address 172.16.0.2/24 --net-gateway 172.16.0.1 k8s/master
vml run --net-tap tap1 --net-address 172.16.0.3/24 --net-gateway 172.16.0.1 k8s/node/1
…
```

После установки пакетов и инициализации мастера можно одной командой подключить

```
vml ssh -p k8s/node -c 'kubeadm join 172.16.0.2:6443 ...'
```

Для остановки всех машин кластера можно запустить команду `vml stop -p k8s`.



ПРЕЗЕНТАЦИЯ
ОКОНЧЕНА

СПАСИБО ЗА
ВНИМАНИЕ :Т

risovach.ru



Алексей Шабалин
Руководитель отдела виртуализации, Базальт СПО
shaba@basealt.ru

https://github.com/Obirvalger/vml

https://github.com/Obirvalger/vml/tree/master/doc





